Today we learned four kinds of data structures.

There are two types of the structures that we have learned today -- linear and non-linear data.

LINEAR:

1. Linked list

In our real world this kind of data structure is just like loose-leaf binder, link chain and treasure hunt game. There are linked allocations of disk space so that there is this file-allocation table for a linked list. Each element in linker list is called a node. A node should contain data and a link which link to the next node. Nodes linked together to form a linked list.

Insert: There is only the need of changing the link. First, link the new node to the next one node

which after the position where the new node will be inserted. Second, link the original node in that position to the new node. A new node has just be inserted quickly.

Delete: To delete a node, it’s just like the process that insert one. It’s also pretty convenient because the only thing that has to be changed is just the link. Link the previous link to the next node will easily delete the element.

1. Stack

This kind of data is just like the tanghulu or lamb shashlik. For any case like this we know that there is only one entrance for anything. For example, there is a stack of clothes. The first cloth we put will stay at the bottom. After we put ten more clothes on the stack, if we are going to get the first cloth we put, all other clothes have to be moved. Data acts in the same way like we putting clothes. There is only one way for the data to get in or get out. This is what we said as “First in, last out.” We can only operate on the top of the stack.

Push(insert): We can only push the data from the top of the stack.

Pop(delete): We can only take out the data on the top of the stack.

1. Queue

A queue looks like we waiting a queue of cars or people. It follows the rule “First in First out”. A queue is going to deal with the data one by one. Don’t be rush, each one has to follow its order. We have to put the data in to a queue from the back and waiting for it to come out from the front of queue.

Enqueue(insert): We have to insert the data at the tail of the queue.

Dequeue(delete):We have to delete the data at the dead of the queue.

NON-LINEAR

1. Tree

The last kind of data structure is the tree. As its name shows, a tree is just like a family tree, personnel tree or biology family tree. Like a plant, there have to be roots and leaves. Each tree will need a root, and from the root there can be as many leaves as we want. There are three kinds of important nodes:root node, internal node, leaf node. A root node is the base and the begin of a tree structure. Then it will play a role of internal node, after comparing with an internal node the new data will be set in the right place as a leaf node. A leaf node will become a internal node as well if there is a new data coming in.

Users’ moves will be based on binary search on an ordered binary tree. The special structure of the tree makes it easy to find a specific data. Inserting and deleting are also based on binary search.